Towards the Alignment of Handwritten Music Scores

Pau Riba, Alicia Fornés, and Josep Lladós

Computer Vision Center - Computer Science Department
Universitat Autònoma de Barcelona, Bellaterra, Catalonia, Spain
{priba,afornes,josep}@cvc.uab.es

Abstract. It is very common to find different versions of the same music work in archives of Opera Theaters. These differences correspond to modifications and annotations from the musicians. From the musicologist point of view, these variations are very interesting and deserve study. This paper explores the alignment of music scores as a tool for automatically detecting the passages that contain such differences. Given the difficulties in the recognition of handwritten music scores, our goal is to align the music scores and at the same time, avoid the recognition of music elements as much as possible. After removing the staff lines, braces and ties, the bar lines are detected. Then, the bar units are described as a whole using the Blurred Shape Model. The bar units alignment is performed by using Dynamic Time Warping. The analysis of the alignment path is used to detect the variations in the music scores. The method has been evaluated on a subset of the CVC-MUSCIMA dataset, showing encouraging results.

Keywords: Optical Music Recognition, Handwritten Music Scores, Dynamic Time Warping alignment

1 Introduction

There are many Opera Theaters worldwide with a huge amount of handwritten music scores. Their archives contain the music scores of the different representations of Operas, Concerts and Ballets. For each one of these representations, many musicians (especially composers and conductors from the 18th-19th centuries) used to slightly modify the original music score with the aim of beautifying, easing the technical difficulties of some parts, etc. As a result, a particular music work could have many different versions due to the differences in the music notes, dynamics, tempo annotations, etc.

Indeed, many scholars focus their research on the analysis of these variations from the musicological point of view. For this purpose, they have to visually compare the different versions of the music composition, which is a time consuming task. Thus, a method that automatically detects the passages that contain variations could undoubtedly reduce their effort.

One solution could be to automatically recognize the handwritten music scores and then compare the resulting MIDI files. Although there are commercial products for Optical Music Recognition (OMR) of printed scores [1], [2], the state of the art in handwritten music recognition [3], [4] is still not mature enough. Nevertheless, there has been a huge advance in the recognition of isolated handwritten music symbols [5], [6] as well as in on-line music recognition [7], [8]. In the case of online OMR, the temporal information can effectively help in the recognition, and even some commercial products exist [9], [10].

However, old handwritten music scores contain degradations, non-standard notation, and huge differences in the musicians' handwriting style. Thus, recognizing the music scores and comparing the MIDI files can not be taken into consideration yet. For this reason, we propose the alignment of music scores as an alternative. The main idea is to compare the two music scores from the visual appearance point of view, avoiding as much as possible its recognition.

In this paper we explore the use of a classical alignment technique for handwritten text and discuss the main difficulties and adaptations that should be taken into account when aligning music scores. The overview of the method is the following. First, braces and ties are removed. Then, music clefs and bar lines are detected. Afterwards, the Dynamic Time Warping (DTW) is used to align the bar units of the two music scores. Finally, the alignment path is analized for two reasons: first to detect the bar units that have been merged, and second, to detect the bar units that have been aligned with a high matching cost. Figure 1 shows the pipeline of the proposed approach.

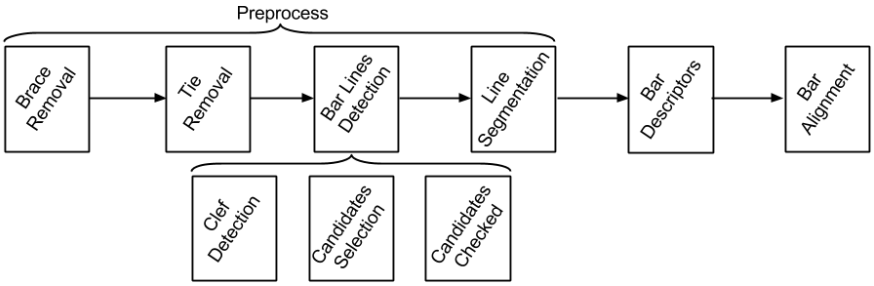


Fig. 1. Overview of the alignment process.

The rest of the paper is organized as follows. Section 2 describes the relevant music elements that have to be detected in the score. Section 3 describes the alignment of the bar units. Section 4 discusses the experimental results. Finally, conclusions and future work are drawn in Section 5.

2 Detection of specific music elements

We assume that the input of our method is a binary image without staff lines. Concerning the staff lines removal, there are many methods in the literature devoted to this task, such as [11], [12]. Indeed, according to the staff removal competitions held at ICDAR [13] and GREC [14], the performance of the current methods is extremely good.

So, the image without staff lines is cleaned by removing small blobs. The next step consists in detecting and removing certain elements that can generate confusion in the segmentation and alignment steps, such as the braces at the beginning of the document and the biggest ties. Then, the bar lines are detected and the different staffs are segmented.

2.1 Brace Removal

Braces appear at the beginning of the document and cover several staffs. They indicate that the music is polyphonic, composed of several staffs that are played in parallel, such as the music scores for piano, quartet, orchestra, choir, etc.

Given that braces can be easily misclassified as bar lines and increase the difficulties in staff segmentation, we propose to detect and delete the braces before the alignment. Thus, the brace detector analyzes the beginning of the staffs and, for each long (height) connected component, it applies a median filter with a vertical structuring element to detect a vertical long line. If the vertical cover several staffs, then it is considered a brace and removed from the image. Note that different music elements may belong to the same connected component. For example, in Figure 9 several music clefs overlap the braces.

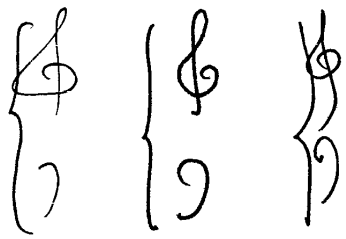


Fig. 2. Braces from different writers. Notice that some treble clefs overlap the braces.

The detailed steps are the following. First, we detect the beginning of the score, crop this region and compute the connected components. For each one of these elements, we must decide whether it is a brace or not. Following the musical notation theory, a brace must cross consecutive staffs. If there are elements satisfying these requirements, they are considered brace candidates. For these candidates, we apply a median filter with a thin vertical window to keep the

almost vertical lines. However, braces might not be completely vertical and the lines that we get from the median filter will be split in to shorter lines. In order to join these lines, a dilatation with a vertical structural element is applied to the median filtered image. That dilation will join the different parts that can belong to the same brace. Finally, the skeleton is computed and each connected component is studied. For each one of these components, they are approximated as a straight line. If the line is long enough then it is considered a brace. Figure 3 shows the detection process of a Brace. The treble clef in this image is not fully connected and only the bottom part is shown.

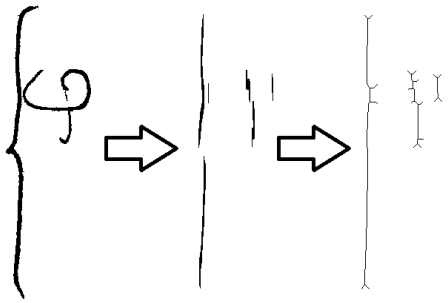


Fig. 3. From left to right: Original connected component, Median filtered image and final skeleton

Once all braces have been detected, they are deleted performing dilatations to the already approximated straight lines. This method is used instead of erasing the whole connected component because we should avoid deleting the clefs that can be next to the brace.

2.2 Tie Removal

Long ties are also common elements. These ties usually cross several bars, increasing the difficulties in line segmentation and bar unit detection (see Figure 4). Also, misclassifying a tie may lead to incorrect music alignments, specially for the largest ones that will propagate these errors to several bar units.

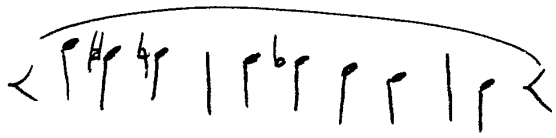


Fig. 4. Tie crossing 3 bars.

For this reason, we detect and remove these long ties by analyzing the aspect ratio of the long (width) connected components. Deleting smaller ones can erase parts of the symbols that are important for the alignment. Figure 5 shows an example where the beam is disconnected from the stems. These cases are easily confused with ties.

The detection step is focused on connected components. We state that a component is a tie only if the aspect ratio and the width are bigger than a predefined threshold. This threshold prevents the algorithm to confuse cases like the ones in the Figure 5 by asking the ties to be long enough.

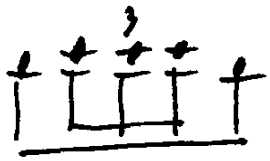


Fig. 5. Beam that can be confused with a tie.

2.3 Clef detection

The detection of clefs at the beginning of each staff can be used to determine when a bar unit starts. When comparing music scores from different writers, it is common to find that a writer compresses the bar units more than others. As a result, the amount of bar units per line is different. In music notation, at the beginning of each staff, the clef and accidentals have to be written again. For this reason, these elements should not be taken into account when comparing the bar units from different writers.

To detect and recognize the clefs, the beginning of each staff is described using the Blurred Shape Model (BSM) [15], which can be considered as a weighed zoning descriptor. The BSM encodes the probability of pixel densities of image regions. The image is divided in a grid of n x m equal-sized subregions. Each bin receives votes from the foreground pixels in the image region, and also from the neighboring bins. In other words, each foreground pixel contributes to the density measure of its bin and its neighboring ones. This contribution is weighted according to the distance to the center of each bin.

For clef detection we follow the same procedure as for brace detection. First, an image region is cropped at the beginning of the staff. Afterwards, a morphological closing is performed to join small elements in the same connected component. If these components are big enough then they are considered a clef candidate. Then the BSM descriptor is computed and compared to a dataset of music clefs from different authors, which is described in [15]. Figure 6 shows some examples of this dataset.

(a) Alto (b) Bass (c) Trebble

Fig. 6. Examples of clefs.

2.4 Bar line extraction

The bar lines are detected with two objectives. On the one hand, they are used to separate the bar units; on the other hand, they can be used to determine the number of voices in the musical score. Moreover, since the alignment is based on comparing the bar units of the music scores, the detection of bar lines is the most important step. A bad detection will lead to serious mistakes in the alignment. For example, Figure 7 shows the problems that can come from the writing style of the author. In that image, it is easy to misclassify the stem of the note with a bar line.

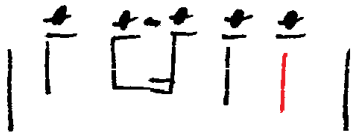


Fig. 7. The last stem (in red) can be confused as a bar line.

Our bar detection consists in the following steps: bar line candidates and candidates test. These steps are described next.

Bar line candidates First, a median filter is applied to detect the lines. Note that a bar line can overlap other elements. Every detected vertical line is marked as a candidate if it (almost) crosses all the staff and does not contain blobs at its extrema points (otherwise it is a note stem). Figure 8 shows the three steps: first the image, secondly the median filter and finally the chosen candidates.

Candidates test We check the consistency of the bar line candidates within their context. For example, we discard a bar line that is much shorter than the others, as well as a bar line that crosses only one staff in a two-voice music score.

For this purpose, we compute the length of all the detected bar lines and discard the outliers. First, we sort the candidates vertically in order to find the different lines of the document. These sorting can be done using the candidates

Fig. 8. Bar lines detection pipeline.

centroid. When the lines have been detected, a set of outliers is computed. We consider that one candidate is an outlier if its length is very different from the candidates in the same line. We check whether some short lines could be vertically joined. Otherwise, they are finally rejected. Afterwards, the lines are checked to be truly separated. If a whole line is completely inside another, it is considered as an error, and all their candidates are deleted. Finally every bar is checked to be wide enough.

We would like to remark that virtual bar lines are added at the beginning and ending of each staff, just in case the musician forgot to draw them.

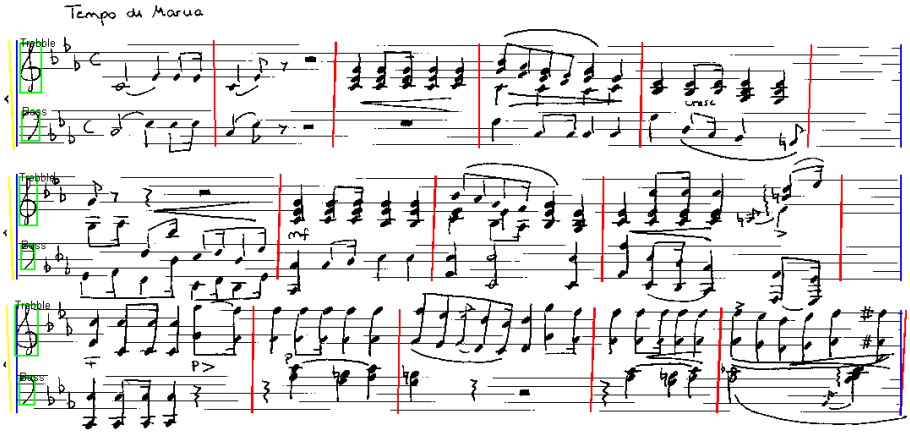


Fig. 9. Detection of braces (in yellow), music clefs (in green) and bar lines (in red). The bar lines in blue color correspond to the virtual bar lines at the beginning or ending of each staff system.

2.5 Staff Segmentation

The next step consists in segmenting the staffs. This is a critical step for elements that appear between the two staffs. For example, Figure 10 shows some

annotations and dynamics that can appear between two staffs. In such cases, the system has to decide which components belong to each staff.

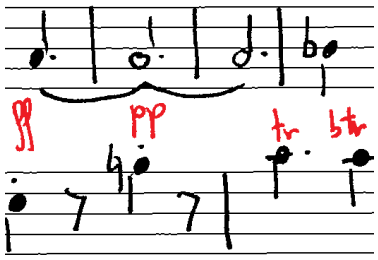


Fig. 10. Line segmentation problems in red color.

First the straight line that crosses through the middle point between staffs is computed. Afterwards, for every connected component that has been cut by that line is analysed. There are two possibilities:

- If the element crosses both staffs, then it is split in two. Thus, the component will be divided by the place of minimum width.
- Otherwise, the element will be assigned to the staff closer to the center of mass of the element.

3 Bar Alignment

The bar alignment can be divided in two stages: bar unit representation and bar alignment. These steps are described next.

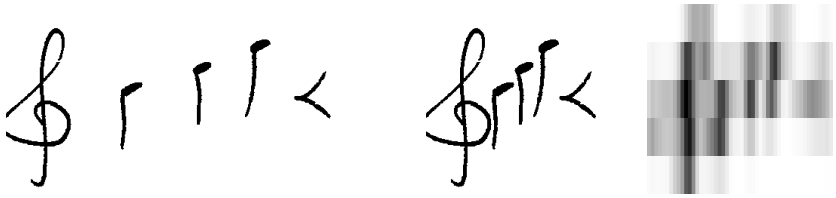
3.1 Bar Unit Representation

For every bar unit, the vertical blank spaces are deleted because the space between notes can vary a lot between different authors. Next, the Blurred Shape Model descriptor is computed. The grid of the descriptor has been empirically set to 5 vertical and 50 horizontal divisions. These steps are shown in Figure 11.

3.2 Bar Unit Alignment

Once we have a descriptor for every bar unit, we can start aligning two music sheets, namely A and B. For this purpose, the Dynamic Time Warping (DTW) with Sakoe-Chiba band algorithm is used [16].

The DTW algorithm was originally proposed by Kruskal and Liberman for putting audio samples into correspondence. DTW is able to warp the time axis



(a) Original (b) No blank (c) Descriptor

Fig. 11. Bar Unit Description.

in order to optimize the best alignment between two signals. In addition, DTW can handle samples of different length avoiding resampling.

Let us define the DTW distance of two time series $C = x_1..x_M$ and $Q = y_1..y_N$ as $DTWCost(C, Q)$ (see Fig. 12(a)). For this purpose, a matrix $D(i, j)$ (where $i = 1..M, j = 1..N$) of distances is computed using dynamic programming:

$$D(i,j) = min \left\{ \begin{array}{l} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{array} \right\} + d2(x_i, y_j) \tag{1}$$

$$d2(x_i, y_j) = x_i - y_j \tag{2}$$

Performing backtracking along the minimum cost index pairs (i,j) starting from (M,N) yields the warping path (Fig. 12(b)). Finally, the matching cost is normalized by the length Z of this warping path:

$$DTWCost(C, Q) = D(M, N)/Z \tag{3}$$

The creation of this path is the most important part of their comparison: it determines which points match (Fig. 12(c)) and are to be used to calculate the distance between the time series.

In case of music alignment, each cell represents the matching cost of the bar units. Since we have a n-dimensional feature vector (in our case, n=5), the aligning cost stored in each cell of the matrix is computed using the square of the Euclidean distance of the BSM descriptors. Formally, if $f_k(a_i)$ corresponds to the k-th feature of the column i of the image A, and $f_k(b_j)$ corresponds to the k-th feature of the column j of the image B, the matching distance $DTWCost(A, B)$ is calculated using the same equations as in Kruskal's method, but instead of the equation 2, the computation of $d2$ will be the sum of the squares of the differences between individual features:

$$d2(x_i, y_j) = \sum_{k=1}^{5} (f_k(a_i) - f_k(b_j))^2 \tag{4}$$

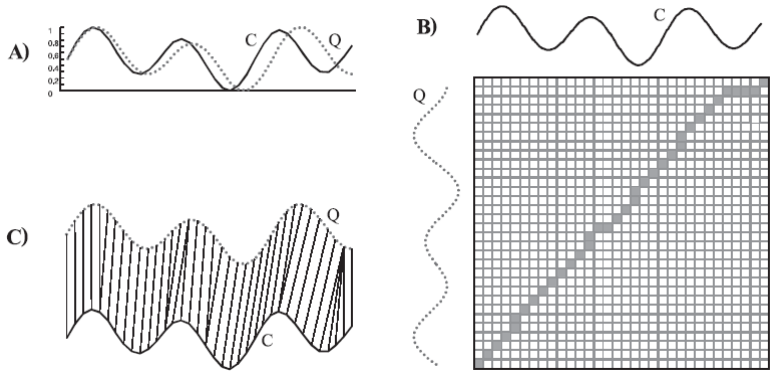


Fig. 12. An example of DTW alignment (extracted from [17]) a) Samples C and Q. b) The matrix D with the optimal warping path in grey color. c) The resulting alignment.

Once the DTW alignment is applied, we focus on the backtracking path along the bars. These values indicate the aligning cost of each pair of bar units. We can consider the backtracking path as a first rough alignment. Then, every problematic alignment is studied. There are two different cases to consider:

- The backtracking path is following a diagonal movement although the matching cost is very high. This case means that the two bar units to be compared are very different (e.g. they contain different music notes). Then, the system marks these two bar units to be shown to the scholar.
- The paths are not following a diagonal movement (i.e. vertical or horizontal movement). It means that one bar unit in A is matching two or more bar units in B or vice-versa. This case can appear due to two reasons:
  - The musician (by mistake or deliberately) added or deleted a bar unit. The system marks these bar units.
  - The system could not detect a bar line (or the musician forgot to draw a bar line). This kind of mistakes have to be avoided as much as possible.

In order to decide whether two bar units should be joined, we compute the mean distance between the bar unit in A and the two bar units in B. Afterwards, we compute the same distance but joining both bar units (this means that we compute the joined BSM descriptor of the two bar units in B). If the second distance is smaller, then we consider that the bar units should be joined and we remove the bar line in the middle. Notice that this decision tends to delete a correct bar line in A whenever a bar line is not detected in B. Contrary, if this second distance is higher, then the algorithm considers that the musician added one extra bar unit and marks this difference.

In this way, every variation between the two music scores is marked and notified to the musicologist.

4 Results

For the experiments, we have selected a subset of the CVC-MUSCIMA dataset [18]. Concretely, we have selected 5 music works written by 8 different musicians (which means that we have 8 different versions). These music scores vary in length, number of voices, etc. For each one of these documents, we have created a ground-truth. Each bar line has an identifier number, corresponding to the order of appearance in the score (see the green numbers in Figure 13).

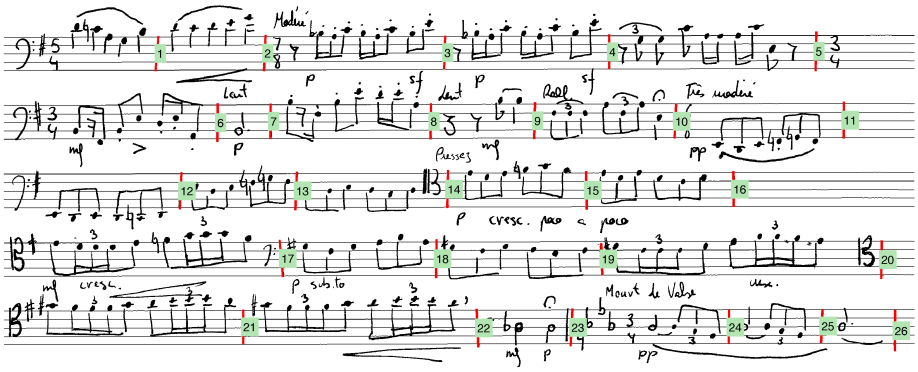


Fig. 13. Ground-truth example.

These identifier numbers are consistent for the different versions. For example, in Figure 14, a musician has merged several bar units (missing one bar line), so the ground-truth indicates this issue by using the correct identifier.

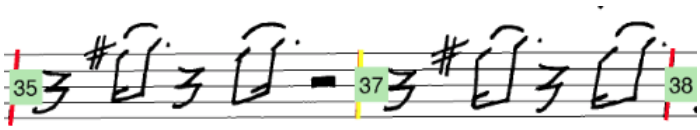


Fig. 14. Ground-truth correction. The musician forgot to write the bar line n.36.

4.1 Bar lines extraction evaluation

The first experiment consists in evaluating the performance of the bar line extraction proposed in section 2.4. It is important to validate it because the alignment highly depends on a correct bar line extraction.

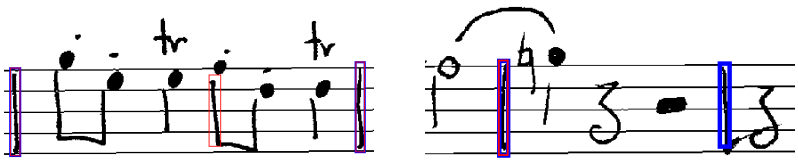
Table 1 shows the performance of the proposed technique. Precision is the fraction of retrieved instances that are relevant and Recall is the fraction of

relevant instances that are retrieved. From the results we can conclude that the bar detection is not perfect although accurate enough for the next stage. It is important to remark that the alignment is able to detect that two bar units should be joined.

Metric	Result
Precision	89.383%
Recall	95.327%

Table 1. Bar Lines Detection

Figure 15a shows an example of False Positive (FP) whereas Figure 15b presents the opposite case, a False Negative (FN).



(a) False Positive (b) False Negative

Fig. 15. Bar descriptor

4.2 Alignment Evaluation

The next experiment evaluates the performance of the alignment approach. This evaluation has been done comparing the bar line identifiers of every pair of aligned music sheets with the ones in the ground-truth. Since there are 8 versions of 5 music works, we have run the alignment 140 times (5 * 28). Table 2 presents the performance of the proposed approach in terms of the number of corrected matched bar units divided by the total amount of bar units in the music score.

Metric	Result
Accuracy	88.743%

Table 2. Alignment results.

Figure 16 shows the same passage from two musicians. Notice that the second writer made a mistake in bar unit 36. In the image, the green numbers correspond

to the identifier in the ground truth, whereas the red ones are the ones proposed by the alignment algorithm. Obviously, the bar unit number 37 is a combination of the 36 and 37 ones. The algorithm detects that there is indeed a variation, marking these bar units and notifying the scholar.

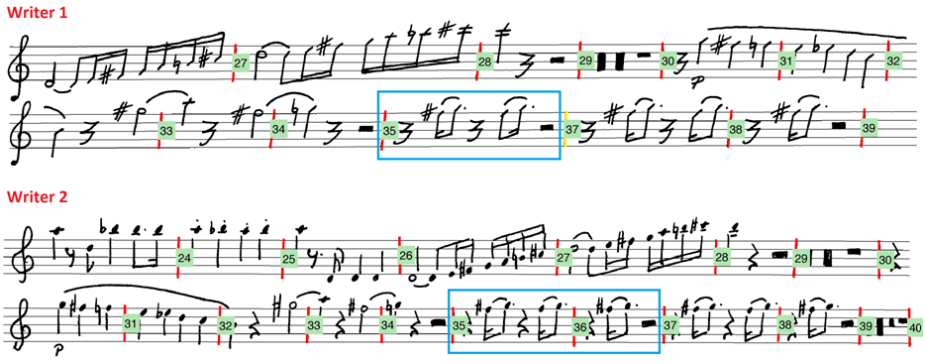


Fig. 16. Detection of a variation between two musical scores.

5 Conclusion and Future Work

In this paper, we have proposed a music score alignment method for detecting variations in music sheets. The method is based on the detection and alignment of bar units using the classical Dynamic Time Warping. We have analyzed the main difficulties and adaptations that must be performed.

The experimental results are encouraging. Thus, we could conclude that music alignment can be seen as a semi-blind tool from the optical music recognition point of view, but with the ability to detect variations between two music scores.

Future work will be focused on analyzing the performance of the bar unit detection. Moreover, we will focus on the detection of variations inside each bar unit. This detection will be done element by element such as notes or dynamics. For this purpose, we plan to use more powerful techniques such as graph-based techniques.

Acknowledgment

This work has been partially supported by the Spanish project TIN2015-70924-C2-2-R, the European project ERC-2010-AdG-20100407-269796 and the *Ramon y Cajal* Fellowship RYC-2014-16831.

References

1. "Photoscore." [Online]. Available: http://www.neuratron.com/photoscore.htm
2. "Sharpeye." [Online]. Available: http://www.visiv.co.uk/
3. A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. Marcal, C. Guedes, and J. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.
4. A. Fornés and G. Sánchez, "Analysis and recognition of music scores," in *Handbook of Document Image Processing and Recognition*. Springer-Verlag London, 2014, pp. 749–774.
5. A. Rebelo, G. Capela, and J. S. Cardoso, "Optical recognition of music symbols: A comparative study," *International Journal on Document Analysis and Recognition*, vol. 13, no. 1, pp. 19–31, 2010.
6. A. Fornés, J. Lladós, G. Sánchez, and D. Karatzas, "Rotation invariant hand drawn symbol recognition based on a dynamic time warping model," *International Journal on Document Analysis and Recognition*, vol. 13, no. 3, pp. 229–241, 2010.
7. H. Miyao and M. Maruyama, "An online handwritten music symbol recognition system," *International Journal of Document Analysis and Recognition (IJDAR)*, vol. 9, no. 1, pp. 49–58, 2007.
8. J. Calvo-Zaragoza, J.; Oncina, "Recognition of pen-based music notation with probabilistic machines," in *Proceedings of the 7th International Workshop on Machine Learning and Music*, Barcelona, Spain, 2014.
9. "Myscript music." [Online]. Available: http://myscript.com/technology/music
10. "Staffpad." [Online]. Available: http://www.staffpad.net/
11. C. Dalitz, M. Droettboom, B. Pranzas, and I. Fujinaga, "A comparative study of staff removal algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 753–766, 2008.
12. J. dos Santos Cardoso, A. Capela, A. Rebelo, C. Guedes, and Pinto, "Staff Detection with Stable Paths," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 1134–1139, 2009.
13. M. Visani, V. C. Kieu, A. Fornés, and N. Journet, "Icdar 2013 music scores competition: Staff removal," in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013, pp. 1407–1411.
14. A. Fornés, A. Dutta, A. Gordo, and J. Lladós, "The 2012 music scores competitions: staff removal and writer identification," in *Graphics Recognition. New Trends and Challenges*. Springer, 2013, pp. 173–186.
15. S. Escalera, A. Fornés, O. Pujol, P. Radeva, G. Sánchez, and J. Lladós, "Blurred Shape Model for binary and grey-level symbol recognition," *Pattern Recognition Letters*, vol. 30, no. 15, pp. 1424–1433, 2009.
16. H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.
17. E. Keogh and C. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, 2005.
18. A. Fornés, A. Dutta, A. Gordo, and J. Lladós, "Cvc-muscima: a ground truth of handwritten music score images for writer identification and staff removal," *International Journal on Document Analysis and Recognition*, vol. 15, no. 3, pp. 243–251, 2012.